

Tracking survey:
Can we use Olympus for sPHENIX case?

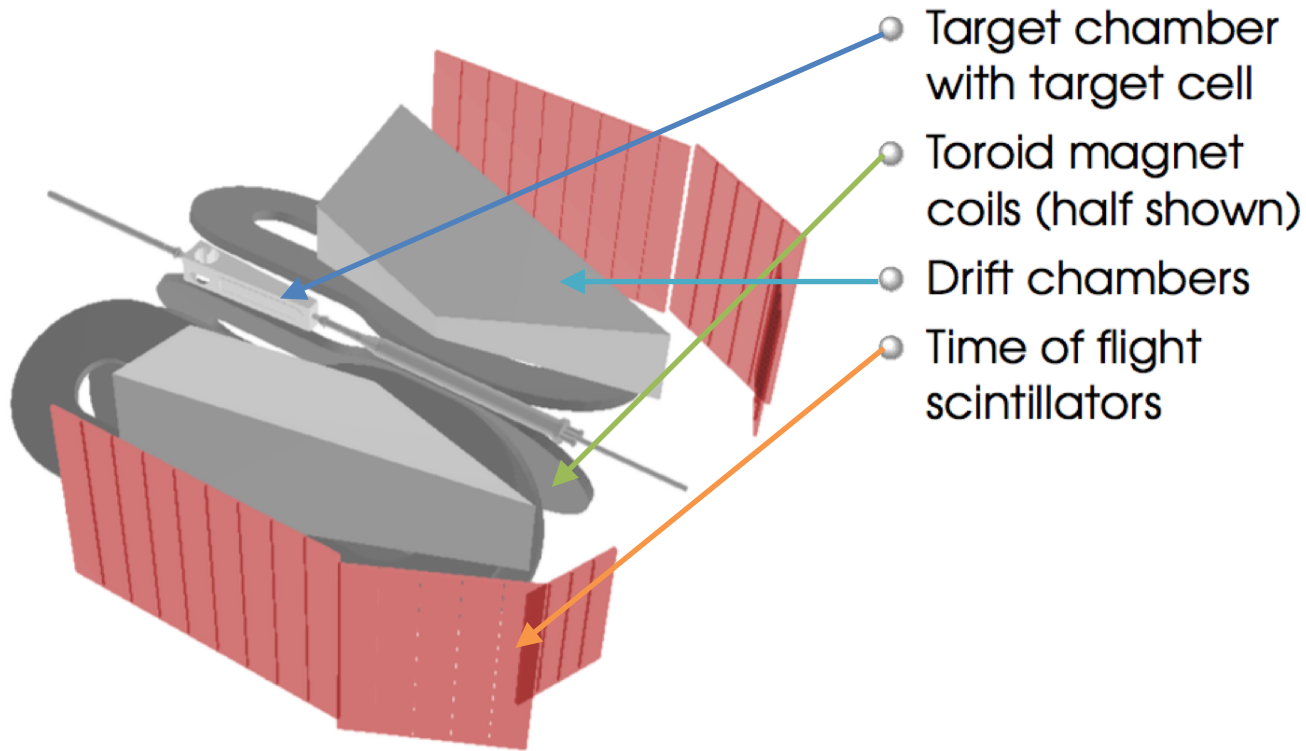
Carlos Perez

Prolog

- During our quest for finding an optimal pattern recognition algorithm, the Olympus software was suggested to us.
- We contacted Jan Bernauer (main developer of Olympus pattern recognition software).
- We had a long and fruitful discussion with Jan last Thursday during TPC meeting where he presented a detailed explanation of the tracking problem they have and how their software attacks it.
- Here I report what the code does and sketch few ideas of how we can use it.

OLYMPUS detector

Anatomy of the OLYMPUS detector

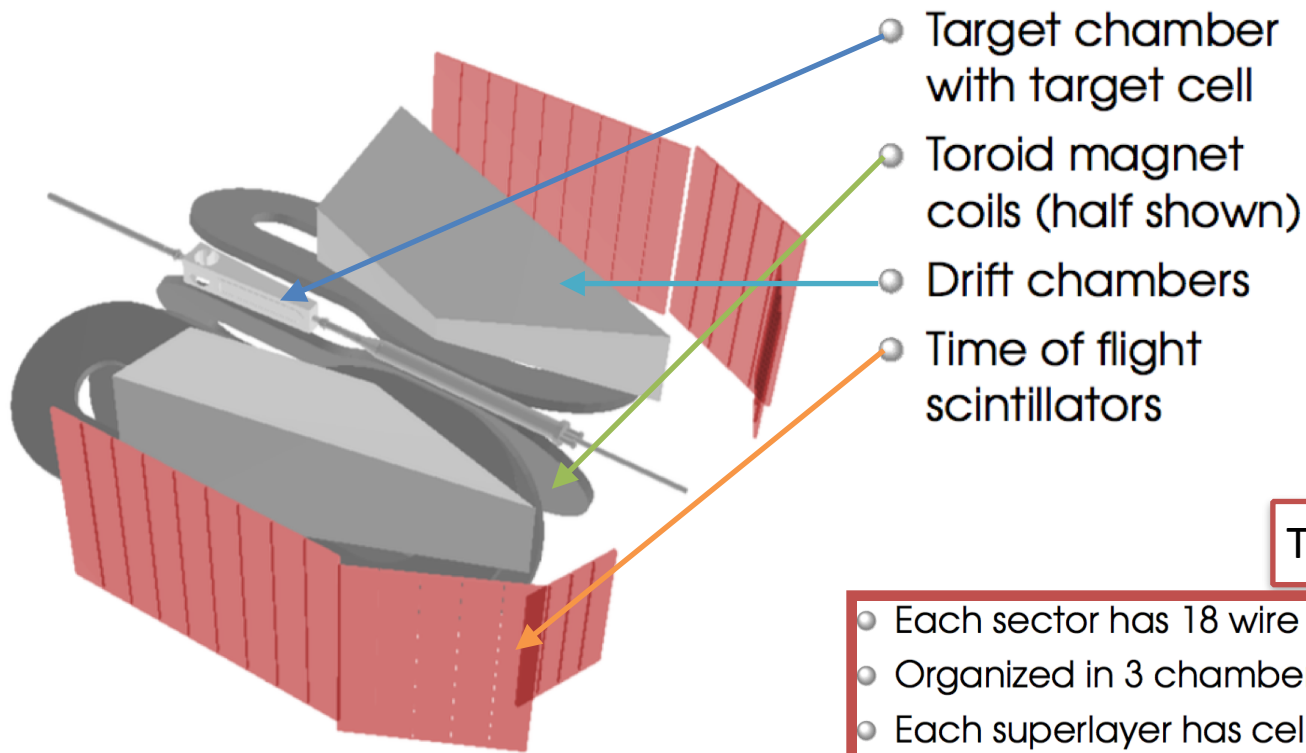


R. Milner et al., NIMA 741 (2014) 1-17

J. Bernauer slide

OLYMPUS detector

Anatomy of the OLYMPUS detector



Tracking MiniStructure

- Each sector has 18 wire layers
- Organized in 3 chambers with 2 superlayers each
- Each superlayer has cells with 3 wires each

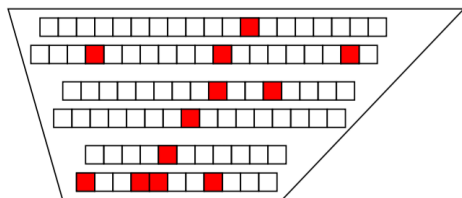
R. Milner et al., NIMA 741 (2014) 1-17

J. Bernauer slide

Pattern Recognition in Olympus

Pattern matching

Problem: Find cells along trajectory within noise hits.



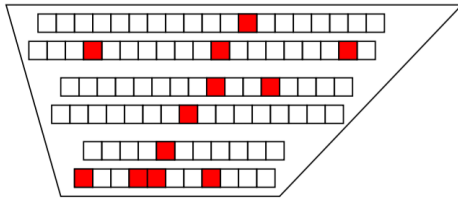
J. Bernauer slide

Pattern Recognition in Olympus

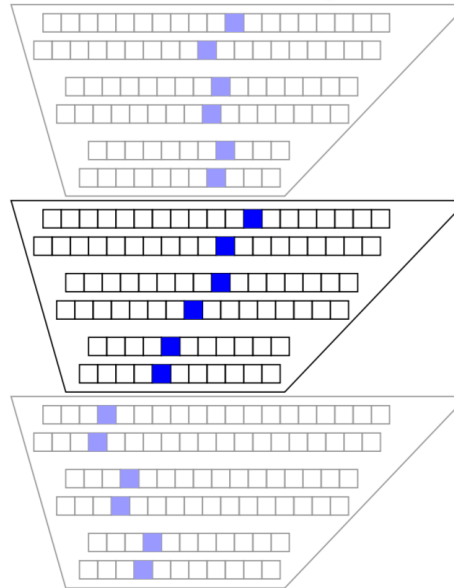
Lib. of templates generated w/MC

Pattern matching

Problem: Find cells along trajectory within noise hits.



Solution: Compare with pattern database.



Advantages:

- Stochastic effects folded in (MC)
- Quick for small grids (like Olympus)
- Larger grids could be reduced (eff)

J. Bernauer slide

Very modular code available

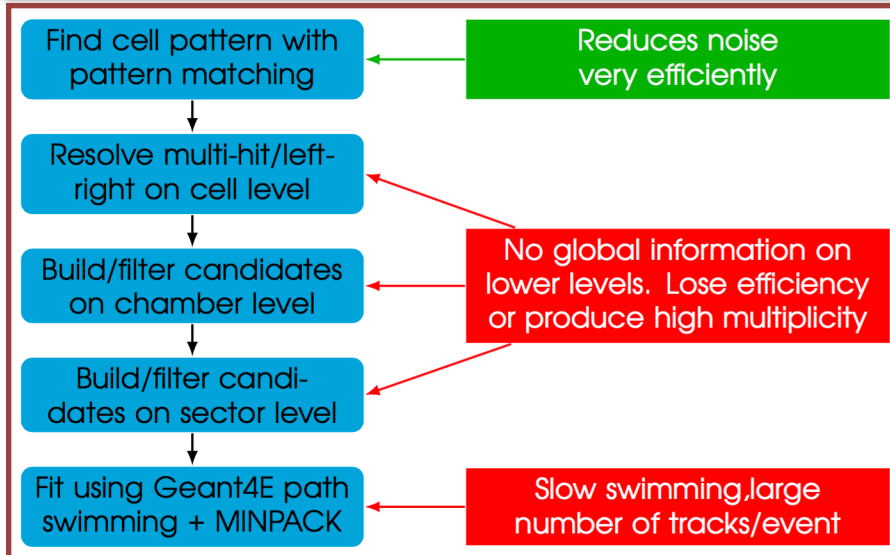
Features

- Templated library, adjustable for any reasonable bit length
- Test if pattern is found
- List all found patterns
- Detector efficiency: can specify number of missed bits
- Available at <https://github.com/JanCBernauer/patternmatch>

How to use

- Split up detector in cells
 - "natural" units like paddels/strips/drift cells
 - arbitrary groups/ranges
- Find all possible patterns (MC, brain, etc.)
- Build database with library
- Profit

Tracking Strategy in a nutshell (standard/old approach for Olympus)



Improved versions of this flowchart are being build for olympus that allows for a faster propagation, less combinatorics and better fitting (splines).

Testing Olympus in sPHENIX...

- After discussion in the TPC group, we decided to test a very minimal implementation of Olympus for our tracking case.
 - e.g. three templates in a standalone run
- Goal is to explore scalability and performance.
- Will report after first tests are completed.